ARL-TR-7410 ● Sep 2015

US Army Research Laboratory

High-Bandwidth Tactical-Network Data Analysis in a High-Performance-Computing (HPC) Environment: Data Marshalling

by Kenneth D Renard, Joseph D Rivera, James R Adametz, and Jordan R Franssen

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

ARL-TR-7410 ● SEP 2015

US Army Research Laboratory

High-Bandwidth Tactical-Network Data Analysis in a High-Performance-Computing (HPC) Environment: Data Marshalling

by Kenneth D Renard
Computational and Information Sciences Directorate, ARL

Joseph D Rivera
Aberdeen Test Center, Aberdeen Proving Ground, MD

Jordan R Franssen and James Adametz
QED Systems, LLC, Aberdeen, MD

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
September 2015

2. REPORT TYPE
Final

3. DATES COVERED (From - To)
July 2012–December 2014

4. TITLE AND SUBTITLE
High-Bandwidth Tactical-Network Data Analysis in a High-Performance-Computing (HPC) Environment: Data Marshalling

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Kenneth D Renard, Joseph D Rivera, James R Adametz, and Jordan R Franssen

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
US Army Research Laboratory
ATTN: RDRL-CIH-C
Aberdeen Proving Ground, MD 21005-5067

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-7410

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Teams from the Aberdeen Test Center and the US Army Research Laboratory collaborated to design and build a system-of-processes with a goal of reliably marshalling data from a large-scale remote tactical network field test, shuttling the data cross-country, and performing reduction and analysis. The progression of going from terabytes of raw collected field data to a data product that can be used by analysts and evaluators required coordination of people and hardware from several cross-country sites. This system-of-processes reliably worked over a 2-month time span, processing an average of 1.5 TB of raw field-collected data a day. This report describes methods, processes, and systems employed to go from raw data to a finished data product in a short time period (several days) to facilitate Army decision making on a major communications system production.

15. SUBJECT TERMS
tactical networks, data reduction, high-performance computing, data analysis, big data

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Kenneth D Renard
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER (Include area code)**
Unclassified	Unclassified	Unclassified	UU	18	410-278-4678

Contents

List of Figures

1. Introduction

The collection and management of data in a fully distributed environment requires the coordination and effort of people across multiple autonomous information processing systems. The complete system must be fault-tolerant and resilient to any issues that may arise. Each of the individual systems involved has its own set of challenges and obstacles to overcome. The experienced gained in previous network testing was vital to the success and overall performance of the final integrated system.

Previous tests allowed the Aberdeen Test Center (ATC) to operate the entire process from collection of field data to a reduced data product as a single unified information system that reached out to remote systems via network border extensions.

In addition to the new complexities involved with integrating different networking groups, the volume of data was expected to dramatically increase in size (between 2 to 5 times previous efforts). This explosion in data volume made utilizing the US Army Research Laboratory's (ARL's) supercomputing resource center a necessity.

After a brief evaluation of our system and processes, it became apparent that a new approach was needed. Our old system was built with a single purpose: to move data from the field to our central data repository. It involved unique end point data aggregation hardware, gave little consideration to an unstable network, and gave no thought to interacting with a high-performance-computing (HPC) infrastructure.

2. New Approach Design Goals

A new approach was needed, one in which the failure of any one component was independent of another. This approach needed to be able to scale across any number of computers, allow for a new process to be inserted into the pipeline, and provide a mechanism to prioritize some parts at the expense of others.

The new data-marshalling design needed to reliably accomplish the 5 major phases of the end-to-end processing:

1. Raw field data are extracted from the data collection devices (harvested), and then that data are transferred from all remote sites back to ATC for archiving and further processing.
2. Metrics extraction and reduction of the raw data. This includes scanning all harvested files for integrity and metadata contents, which will help to ensure

that the HPC system has access to all required raw data. Once all data files are available, the system will launch the HPC jobs to render the raw data into an initial set of data products.

3. Loading the HPC results into a query able data model. This includes pulling the initial data products back from the HPC domain, creating an empty data model on an ATC database server, and loading the HPC-derived data into an unauthenticated data model.
4. Render quality assurance reports on the raw and reduced data. These reports will then be used by the data authentication team to determine how to mark the data for analytical use.
5. Once this is done the data authentication team will provide manual inputs to a data-marking process. The marked-up database is then considered to be "authenticated" and ready for analysts and evaluators to use.

The following Figure depicts both the concept of operations (CONOPS) and the overall process flow as data progresses through the new data marshalling design plan.

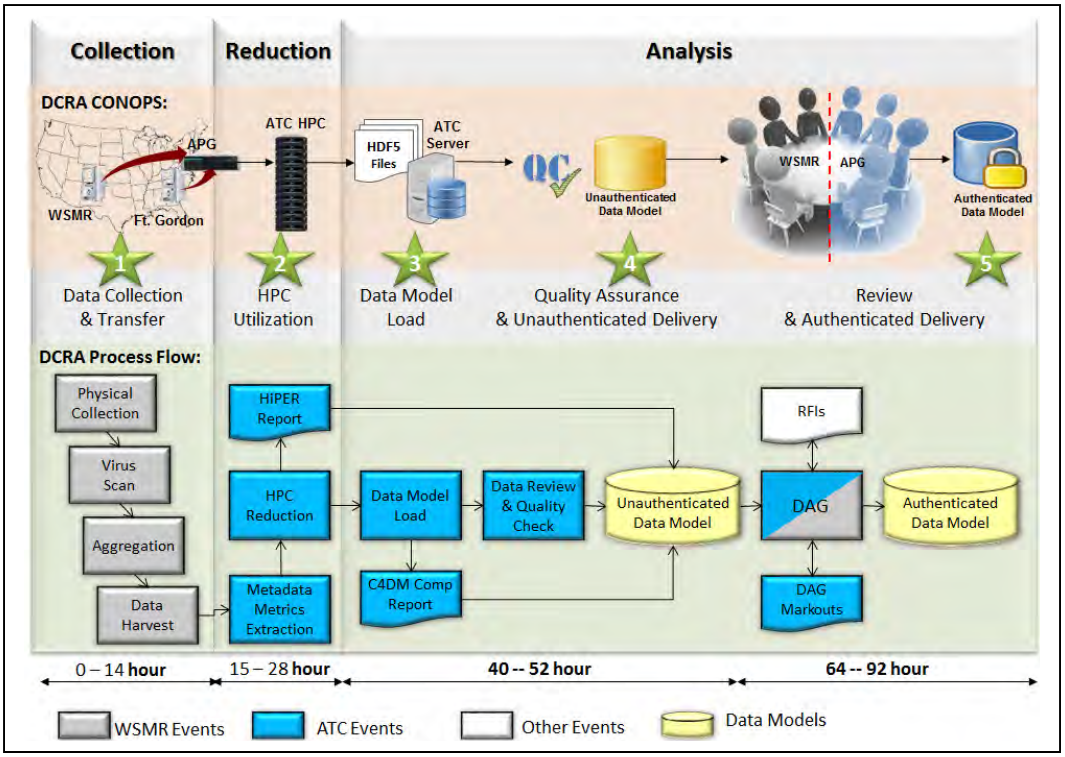

Figure. Data collection reduction and analysis process flow and CONOPS

Once this system has retrieved all harvested test data and synchronized it with the HPC, it will join with the field-collected contextual data. Field context data are data describing the test and are critical for understanding where the data came from. The process of correlating and aggregating it all together is then performed. This system uses software that breaks the reduction down into smaller, parallel components—a prime factor to enable the system to scale the processing as the size of the data grew.

After the HPC processing is complete, the results must be retrieved. The process management infrastructure job is watched for a successful completion. The results are then pulled back from the HPC in their native format. This format is not easily worked on by a general analyst and must be converted and loaded into a standard database capable of being queried with SQL.

With this new design implemented by both the ARL and ATC teams, the system was able to achieve our timeline goal of creating a data-deliverable in about 72 h, even as the data ballooned to more than 1.5 TB a day, while remaining resilient in a shared operational information system environment, which included power outages, network outages, and network slowdowns.

The creation of this dynamic and fully distributed system was driven by the need to support testing during Network Integration Exercise 15.1. The tactical network communications analyzed were carried over both terrestrial- and satellite-based systems.

3. Instrumentation

Collection of test data for analysis is carried out by a variety of instruments in a number of configurations. One of the critical pieces of instrumentation is the Advanced Distributed Modular Acquisition System (ADMAS). The ADMAS is a fully customizable instrumentation platform.[1] ATC leveraged the modular nature of the ADMAS and built a version capable of network packet collection. This network-configured ADMAS is a passive collector; it merely observes traffic. The other critical collection device is the active poller, named Hydra. Hydra is attached to the tactical network and actively queries different devices for test configurable pieces of data.

Each of these instruments is attached to specific nodes on the network or configuration items (CIs). Because of the sheer size and complexity of the network, instrumenting every CI is unrealistic. Selection of CIs to instrument and placement within the network topology is carefully considered during test planning. The result is a distributed packet collection and data polling network that, once combined, gives a clear view of the network.

4. Data Harvesting

There are 5 basic phases to the end-to-end marshalling of the raw data to a final data product. These are denoted in the previous Figure by green stars. During phase 1, data are collected on each one of the systems-under-test onto a removable hard drive. Once a day during test execution, the instruments are shut off, the drives full of data are removed, and fresh empty drives are inserted. The instruments are then powered on and collection is resumed. The removed drives are delivered to network harvest kiosks.

After all the drives have been collected and accounted for, they are scanned for viruses and then loaded into the harvest computers. These harvest computers are general-purpose computers with simple software to copy all of the data to their own drives. Once all of the data have been copied off the original drive, it is marked as harvested by the system software, and the drive is purged by the harvest operator in preparation for the drive to return to service.

5. Data Transport

Phase 1 of the system processing ends with the transporting of the harvested data back to the central processing facility at ATC. During the entire harvest process, the central process management infrastructure located at ATC is continuously probing the end unit harvesters searching their local storage for harvested files that need to be retrieved. Once located, a cascade of actions are set in motion. Each and every file is added to a processing pipeline. This pipeline manages the file's existence, starting on the harvest machine, to its final resting place on the HPC.

All phases of the entire data marshalling process utilize Advanced Message Queuing Protocol.[2] Each action that must be applied to a data file is passed as a message to the central broker. Workers are connected to the broker, consume these messages, and perform the designated task. If a network or system fails, the message is reinserted into the queue and retried at a later time, thus making the system more fault tolerant.

6. Raw Data Metrics Extraction

Every file has a simple workflow. The file is first detected on a harvester, then retrieved from across the network, and a copy is made. Once the system has 2 copies of the file (a working copy and an archive copy), a record of the file's location is created in a File Metadata database. The file is also scanned for integrity and rudimentary repair operations are performed if errors are detected. Once a wellformed file is obtained, basic metrics used by the field technicians are extracted and added to the files record in the database. The following are a few examples of the metrics extracted: file size, creation date, harvest location, data collector ID, and file sequence number. Upon completion of metrics extraction, the file is copied to its final stop on the HPC where phase 2 of the process is run: the reduction of the raw data utilizing the ARL HPC facilities.

7. Context Data

In order for any meaningful statistics to be drawn from the harvested data, context information is required. Context information is that which identifies the network topology and the physical location of data collectors and network devices.

Before a test event begins, context information, such as Internet Protocol and media access control addresses, the pairing of instrumentation to CI, and ADMAS collection points, is determined and provided to ATC Command, Control, Communications, and Computers (C4) Analysis to be used as a baseline for context information during the test. During test execution, daily updates are provided to account for changes to instrumentation and the CIs. The Metadata database is used to store all Context data according to their dates of applicability, ensuring no historical information is lost during an update.

During phase 2, the Context data are extracted from the Metadata database based on the time window that is desired for analysis. An HPC run configuration is then automatically generated to match the time window. This configuration specifies exactly what software components to run. The configuration file also has a list of every harvested file that is applicable to the run. The configuration and the context are then copied over to the HPC, and a job is added to the HPC processing queue, where it waits for compute resources to become available to execute its task. This can take minutes to hours depending on the HPC load at the time. An AMQP message is then generated to monitor the job state.

8. HPC Processing

Once the HPC job has allocated all the requested resources (typically 512 or 1024 processing cores depending on the job size), the system runs a script on each of the processes. This script begins by preparing the environment. It first loads the job queuing system and the implementation of Message Passing Interface[3] for the corresponding hardware. This is followed by loading a preconfigured local environment used to bootstrap virtual environments. This bootstrap environment contains “virtualenv”[4] and some convenience wrappers. All reduction framework software dependencies are contained in a virtualenv, simplifying software deployment across the HPC resources. Containing the dependencies in this manner creates a way to isolate the environment. It also provides a way to update and test specific dependencies in an environment that will not affect the rest of the system's users. For example, a new processing library or version of Python can be used without affecting other operations

Once the environment has finished loading, the script creates a specific output directory for the new job. It starts the reduction by calling "mpirun", which spawns multiple instances of the reduction software framework. The reduction framework then reads the configuration and goes through the map-reduce style phases. Once the data processing is complete, the script modifies all output data file permissions for group access. The script will create a "Failed" or "Success" file to indicate whether or not the job completed successfully (this is used as a signal to the AMQP poller that is waiting for job completion). It will then deactivate the virtual environment, and the job completes.

The entire time the job is submitted for reduction on the HPC, it is monitored by the the process management framework (AMQP back at ATC). Approximately every 5 min, an AMQP process accesses the run status of the HPC jobs. If the job is pending start-up or is still in progress, the message is recycled and tried again in another 5 min. If the run has failed, an error is reported and the message is dropped. If the run has finished successfully, phase 3 of the processing will begin with a message being dispatched for each one of the output files to be retrieved. Upon successful retrieval of all output files, a message is dispatched to convert the files and load a C4 Data Model (C4_DM) database.[5]

10. An Analytical Database Product

The files that are generated by the HPC reduction include rendered Voice call audio files (WAV), network packet capture files (PCAP), SNMP dump files (CSV), and Hierarchical Data Format V5 (HDF5) files.[6] These last files contain the primary source of information used by the analysts. To provide a user-friendly interface and to facilitate a multiuser environment for analysts, the HDF5 files are converted into a queryable database following the data model design in the C4_DM.

An ATC C4 analysis-developed tool called Xfer is used to read the HDF5 outputs of the HPC reduction and load the data into a PostGreSQL database.[7] The program uses configuration files to determine the set of tables that need to be loaded and how the members of the HDF files map to database tables and columns.

Xfer breaks down the procedure of loading a database into small, near-atomic tasks, including the creation of a blank database, the read and load of each file, and any required postprocessing actions. Like the HPC reduction, Xfer was created to modularize its tasks and execute them across many processes in parallel. This greatly reduces the amount of time required to migrate the outputs of the HPC to a database that can be queried.

Phase 4 of the process requires manual intervention. It starts with the unauthenticated data model produced in phase 3. A team of analysts, evaluators, and test engineers (commonly called the Data Authentication Group, or DAG) will regularly meet to discuss the quality aspects of the data in the data model. Inputs to these DAG meetings include 2 reports that are derived from HPC processing outputs. The first is known as the HiPER report (High-Performance Evaluation of Raw data). This report shows the overall quality of the coverage of the data collected in time, the accuracy of the time-tagging of the data, and GPS location quality (just to name a few reported items). The second report used by the DAG is the C4_DM Composition report. This report contains charts that inform overall coverage of each network node with respect to time in the analytical data model, with the main focus being the quality of the inter-node communications.

11. DAG Marking and Data Model Release

The primary goal of the DAG meetings is to recommend how to mark the data in the C4_DM before it is released for analysis. These markings include indicators to the analysts as to how certain records can/cannot be used. For instance, if a data collector had a gap in collection, then completion rates to/from the network node that collector was in cannot be used for completion rates. The full set of limited use and reason codes are documented in an ATC report.[5] Phase 5 (see the Figure in Section 2) takes the DAG-generated markout criteria and applies it to the unauthenticated data model. The result is an authenticated data model that can now be used by the analysts and system evaluators to render reports on the system under test.

12. Conclusion

Many of the analysts who used the output of the system-of-processes described in this report work in the Aberdeen Proving Ground, MD, or Washington, DC, areas. Having access to the data products produced by this system near most of the analysts saved time and money by not having the analysts travel to the remote testing facilities.

The amount of data transferred and processed daily exceeded 1.5 TB. Only through using automated processes and the power in the ARL-HPC systems was the team able to accomplish the task of producing a relevant data model in the time allotted.

This system was the culmination of several years of hard work. Many different people with varying backgrounds had to come together to create this dynamic, scalable, and resilient system.

13. References

1. Banks GQ. The modular instrumentation family; 2015 Jan–Feb [accessed 2015 Feb 15]. http://www.dau.mil/publications/DefenseATL/DATLFiles/Jan-Feb2015/Banks.pdf.

2. Technical commitee: OASIS advanced message queuing protocol (AMQP). OASIS advanced message queuing protocol; 2012 Oct 29 [accessed 2015 Jan 21]. http://docs.oasis-open.org/amqp/core/v1.0/amqp-core-complete-v1.0.pdf.

3. Barney B. Message passing interface (MPI). Livermore (CA): Lawrence Livermore National Laboratory; 2014 Dec 15 [accessed 2015 Jan 15]. https://computing .llnl.gov/tutorials/mpi/.

4. Virtualenv. The open planning project; 2015 Jan 11 [accessed 2015 Jan 15]. https://virtualenv.pypa.io/en/latest/.

5. Adametz J. Army Test Center, Analysis. C4 data model description document 1.8.13. Aberdeen Proving Ground (MD): Army Research Laboratory (US); Aberdeen Test Center; not yet published.

6. The HDF group. What is HDF5? Champaign (IL): The HDF group; 2011 May 16 [accessed 2015 Jan 15]. http://www.hdfgroup.org/HDF5/whatishdf5.html.

7. The PostgreSQL Global Development Group. PostgreSQL 9.2.9 documentation [accessed 2015 Jan 15]. http://www.postgresql.org/docs/9.2/static/tutorial.html.

List of Symbols, Abbreviations, and Acronyms

ADMAS	Advanced Distributed Modular Acquisition System
AMQP	Advanced Message Queuing Protocol
ARL	US Army Research Laboratory
ATC	US Army Aberdeen Test Center
CI	configuration items
CONOPS	concept of operations
DAG	Data Authentication Group
HDF 5	Hierarchical Data Format V5
HPC	high-performance computing

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
2 (PDF)	DIRECTOR US ARMY RESEARCH LAB RDRL CIO LL IMAL HRA MAIL & RECORDS MGMT
1 (PDF)	GOVT PRINTG OFC A MALHOTRA
1 (PDF)	DIR USARL RDRL CIH C K RENARD

INTENTIONALLY LEFT BLANK